


Terms used **shutdown matrix error fault failure**

Found 115 of 145,519

Sort results by

Display results

 [Save results to a Binder](#)

 [Search Tips](#)

☐ Open results in a new window

Try an [Advanced Search](#)

Try this search in [The ACM Guide](#)

Results 1 - 20 of 115

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [next](#)

Relevance scale ☐ ☐ ☐ ☐ ☐

1 [Survey of software tools for evaluating reliability, availability, and serviceability](#)

Allen M. Johnson, Miroslaw Malek

September 1988 **ACM Computing Surveys (CSUR)**, Volume 20 Issue 4

Full text available:  [pdf\(3.79 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

In computer design, it is essential to know the effectiveness of different design options in improving performance and dependability. Various software tools have been created to evaluate these parameters, applying both analytic and simulation techniques, and this paper reviews those related primarily to reliability, availability, and serviceability. The purpose, type of models used, type of systems modeled, inputs, and outputs are given for each package. Examples of some of the key modeling ...

2 [Hardware fault containment in scalable shared-memory multiprocessors](#)

Dan Teodosiu, Joel Baxter, Kinshuk Govil, John Chapin, Mendel Rosenblum, Mark Horowitz

May 1997 **ACM SIGARCH Computer Architecture News , Proceedings of the 24th annual international symposium on Computer architecture**, Volume 25 Issue 2

Full text available:  [pdf\(2.05 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Current shared-memory multiprocessors are inherently vulnerable to faults: any significant hardware or system software fault causes the entire system to fail. Unless provisions are made to limit the impact of faults, users will perceive a decrease in reliability when they entrust their applications to larger machines. This paper shows that fault containment techniques can be effectively applied to scalable shared-memory multiprocessors to reduce the reliability problems created by increased mach ...

3 [MPICH-V: toward a scalable fault tolerant MPI for volatile nodes](#)

George Bosilca, Aurelien Bouteiller, Franck Cappello, Samir Djilali, Gilles Fedak, Cecile Germain, Thomas Herault, Pierre Lemarinier, Oleg Lodygensky, Frederic Magniette, Vincent Neri, Anton Selikhov

November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing**

Full text available:  [pdf\(204.28 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Global Computing platforms, large scale clusters and future TeraGRID systems gather thousands of nodes for computing parallel scientific applications. At this scale, node failures or disconnections are frequent events. This Volatility reduces the MTBF of the whole system in the range of hours or minutes. We present

MPICH-V, an automatic Volatility tolerant MPI environment based on uncoordinated checkpoint/rollback and distributed message logging. MPICH-V architecture relies on Channel Memories, C ...

4 Fast detection of communication patterns in distributed executions

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available:  [pdf\(4.21 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

5 Design and test strategies for a safety-critical embedded executive

Charles A. Meyer, Michael G. Reznick

December 1996 **Proceedings of the conference on TRI-Ada '96: disciplined software development with Ada**

Full text available:  [pdf\(900.36 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

6 An introduction to fault tolerant parallel simulation with EcliPSe

Felipe Knop, Edward Mascarenhas, Vernon Rego, V. S. Sunderam


December 1994 **Proceedings of the 26th conference on Winter simulation**

Full text available:  [pdf\(833.92 KB\)](#) Additional Information: [full citation](#), [references](#), [index terms](#)

7 Multiprocessor self diagnosis, surgery, and recovery in air terminal traffic control

W. Walther

January 1973 **ACM SIGOPS Operating Systems Review , Proceedings of the fourth ACM symposium on Operating system principles**, Volume 7 Issue 4


Full text available:  [pdf\(533.10 KB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

The rapid growth of global aviation for business and pleasure has created the need for automated terminal systems of increasing complexity and capability. Continued increases in the aircraft population will require higher levels of automation. Sperry Univac is responding to this challenge with a multiprocessing system, including hardware and software, currently under development which will enable controllers to safely manage the crowded skies.

8 Industrial/government track: The data mining approach to automated software testing

Mark Last, Menahem Friedman, Abraham Kandel

August 2003 **Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining**

Full text available:  [pdf\(296.40 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

In today's industry, the design of software tests is mostly based on the testers' expertise, while test automation tools are limited to execution of pre-planned tests only. Evaluation of test outputs is also associated with a considerable effort by human testers who often have imperfect knowledge of the requirements


specification. Not surprisingly, this manual approach to software testing results in heavy losses to the world's economy. The costs of the so-called "catastrophic" software failures ...

Keywords: automated software testing, finite element solver, info-fuzzy networks, input-output analysis, regression testing

9 Risks to the public: Risks to the public in computers and related systems

Peter G. Neumann


May 2002 **ACM SIGSOFT Software Engineering Notes**, Volume 27 Issue 3

Full text available:  [pdf\(1.92 MB\)](#) Additional Information: [full citation](#)

10 State space exploration in Markov models

Edmundo de Souza e Silva, Pedro Mejiá Ochoa

June 1992 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1992 ACM SIGMETRICS joint international conference on Measurement and modeling of computer systems**, Volume 20 Issue 1


Full text available:  [pdf\(1.35 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Performance and dependability analysis is usually based on Markov models. One of the main problems faced by the analyst is the large state space cardinality of the Markov chain associated with the model, which precludes not only the model solution, but also the generation of the transition rate matrix. However, in many real system models, most of the probability mass is concentrated in a small number of states in comparison with the whole state space. Therefore, performability measures may ...

11 ARIES: a transaction recovery method supporting fine-granularity locking and partial rollbacks using write-ahead logging

C. Mohan, Don Haderle, Bruce Lindsay, Hamid Pirahesh, Peter Schwarz

March 1992 **ACM Transactions on Database Systems (TODS)**, Volume 17 Issue 1

Full text available:  [pdf\(5.23 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)


DB2TM, IMS, and TandemTM systems. ARIES is applicable not only to database management systems but also to persistent object-oriented languages, recoverable file systems and transaction-based operating systems. ARIES has been implemented, to varying degrees, in IBM's OS/2TM Extended Edition Database Manager, DB2, Workstation Data Save Facility/VM, Starburst and QuickSilver, and in the University of Wisconsin's EXODUS and Gamma d ...

Keywords: buffer management, latching, locking, space management, write-ahead logging

12 Columns: Risks to the public in computers and related systems

Peter G. Neumann

March 2004 **ACM SIGSOFT Software Engineering Notes**, Volume 29 Issue 2

Full text available:  [pdf\(165.39 KB\)](#) Additional Information: [full citation](#)

13 Protection and the control of information sharing in multics

Jerome H. Saltzer

July 1974 **Communications of the ACM**, Volume 17 Issue 7

Full text available:  [pdf\(1.75 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


The design of mechanisms to control the sharing of information in the Multics system is described. Five design principles help provide insight into the tradeoffs among different possible designs. The key mechanisms described include access control lists, hierarchical control of access specifications, identification and authentication of users, and primary memory protection. The paper ends with a discussion of several known weaknesses in the current protection mechanism design.

Keywords: Multics, access control, authentication, computer utilities, descriptors, privacy, proprietary programs, protected subsystems, protection, security, time-sharing systems, virtual memory

14 The performance of a service for network-aware applications

Katia Obraczka, Grig Gheorghiu

August 1998 **Proceedings of the SIGMETRICS symposium on Parallel and distributed tools**


Full text available:  pdf(984.29 KB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

15 Power minimization in IC design: principles and applications

Massoud Pedram

January 1996 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 1 Issue 1

Full text available:  pdf(550.02 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


Low power has emerged as a principal theme in today's electronics industry. The need for low power has caused a major paradigm shift in which power dissipation is as important as performance and area. This article presents an in-depth survey of CAD methodologies and techniques for designing low power digital CMOS circuits and systems and describes the many issues facing designers at architectural, logical, and physical levels of design abstraction. It reviews some of the techniques and tool ...

Keywords: CMOS circuits, adiabatic circuits, computer-aided design of VLSI, dynamic power dissipation, energy-delay product, gated clocks, layout, low power layout, low power synthesis, lower-power design, power analysis and estimation, power management, power minimization and management, probabilistic analysis, silicon-on-insulator technology, statistical sampling, switched capacitance, switching activity, symbolic simulation, synthesis, system design

16 Deferred Execution: An "ACE" of an application

Donald A. Link, Martin W. Gardner

May 1979 **ACM SIGAPL APL Quote Quad , Proceedings of the international conference on APL: part 1**, Volume 9 Issue 4

Full text available:  pdf(631.04 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Deferred Execution is an APL application that provides users with a "batch" APL facility. This application was made possible by the design and implementation of ACE (Automatic Control of Execution). The motivation behind ACE and a summary of its facilities are provided. The paper also includes the design goals for a Deferred-Execution system and summarizes the final design used. Finally an exciting new approach to application structure is presented, showing how Deferred Executio ...


17 Summary of the sigmetrics symposium on parallel and distributed processing


Jeffrey K. Hillingsworth, Barton P. Miller

March 1999 **ACM SIGMETRICS Performance Evaluation Review**, Volume 26 Issue

18 Features: Leveraging Application Frameworks

Douglas C Schmidt, Aniruddha Gokhale, Balachandran Natarajan
July 2004 **Queue**, Volume 2 Issue 5

Full text available:  [pdf\(1.60 MB\)](#)

 [html\(38.98](#)

[KB\)](#)

Additional Information: [full citation](#)

19 Curriculum 68: Recommendations for academic programs in computer science: a report of the ACM curriculum committee on computer science


William F. Atchison, Samuel D. Conte, John W. Hamblen, Thomas E. Hull, Thomas A. Keenan, William B. Kehl, Edward J. McCluskey, Silvio O. Navarro, Werner C. Rheinboldt, Earl J. Schweppe, William Viavant, David M. Young
March 1968 **Communications of the ACM**, Volume 11 Issue 3

Full text available:  [pdf\(6.63 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#)

Keywords: computer science academic programs, computer science bibliographies, computer science courses, computer science curriculum, computer science education, computer science graduate programs, computer science undergraduate programs

20 Automatically characterizing large scale program behavior

Timothy Sherwood, Erez Perelman, Greg Hamerly, Brad Calder
October 2002 **Proceedings of the 10th international conference on Architectural support for programming languages and operating systems**,
Volume 30 , 36 , 37 Issue 5 , 5 , 10

Full text available:  [pdf\(1.54 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Understanding program behavior is at the foundation of computer architecture and program optimization. Many programs have wildly different behavior on even the very largest of scales (over the complete execution of the program). This realization has ramifications for many architectural and compiler techniques, from thread scheduling, to feedback directed optimizations, to the way programs are simulated. However, in order to take advantage of time-varying behavior, we must first develop the analy ...

Results 1 - 20 of 115

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)